ECE 415 Control Systems, Fall 2020
Laboratory 1

Objective: To learn some of the useful control design tools provided by MATLAB and to put into practice the concepts seen in class.

Deliverable: A written report containing:

a) For each step, the corresponding block diagrams and any other analytic design tools used, including derivation of transfer functions for open and closed loop designs (in other words, provide an *explanation* for the MATLAB code given in Part I, and for your own control design in Part II).
b) Plots obtained at each step.
c) *In Part II only*, the MATLAB code used at each step to solve the given problem.

The report should be typewritten, well presented (presentation counts towards the grade), and in good English. *No hand-written report will be accepted* (except, perhaps, for diagrams). ***Submit it in PDF format via Isidore.***

It is imperative that you strictly adhere to the honor code. This Laboratory will account for 35% of the Projects category.

Part I

In this first part, a motor speed problem is presented and its solution is given step by step, together with the corresponding MATLAB instructions. Please follow the derivation and implement the given code, and, where asked to, provide a justification for what the code does using block diagrams or mathematical analysis.

0) Consider the DC motor described by the simplified first order differential equation

$$\frac{dy}{dt} + 60y = 600u - 1500w$$

where y is the motor speed, u is the input armature voltage, and w is a load (a disturbance, from the control design point of view).

Assume the initial conditions are zero, and take the Laplace transform:

$$Y(s)(s+60) = 600U(s) - 1500W(s)$$

$$Y(s) = \frac{600}{s+60}U(s) - \frac{1500}{s+60}W(s)$$

We will design open and closed loop **proportional (P)** controllers for this plant. The control objective is to minimize steady-state error and provide good disturbance rejection capabilities.

Open-loop proportional control

The controller is designed by assuming that $w = 0$ (that is, no external load is applied on the motor). From the final value theorem, we set the controller gain to

$K = \frac{60}{600}$ (justify this choice!).

1) Find the impulse response of the open-loop control system.

```
% Define the plant: DC motor

num_motor = 600;
den_motor = [1 60];

motor = tf(num_motor, den_motor)

% Open-loop controller

K = 60/600;
ol_cont = tf(K,1)

% Compute the transfer function from reference to output, without
disturbance

undist_plant = series(ol_cont, motor)

% Plot impulse response

impulse(undist_plant);
title('Open-loop control')
```

2) Plot the undisturbed step response, for a step of magnitude 100.

```
% Plot response to a step of size 100

step(100*undist_plant);
title('Open-loop response to a step of size 100')
```

3) Plot the response to a unit step disturbance, setting the reference to zero. Give the block diagram of what you are doing here and in the previous item.

```
% Compute transfer function from disturbance to output, without
reference

dist_plant = (-1500/600)*motor

% Plot response to a unit step disturbance

step(dist_plant);
```

```
title('Open-loop response to a unit step disturbance')
```

4) Plot the open-loop response when both reference and disturbance are present.

```
% Compute response to both reference and disturbance

t = [0:0.001:0.1];                    % time vector
y_ref = step(100*undist_plant,t);     % response due to reference
y_dist = step(dist_plant,t);          % response due to disturbance

y = y_ref+y_dist;                     % total response (applying
                               %superposition!)

plot(t,y);
xlabel('Time')
ylabel('Motor speed')
title('Open-loop response to a step reference of magnitude 100 and a
unit step disturbance');

% Put undisturbed and disturbed responses in one plot

hold on;
plot(t,y_ref,'--');
legend('With disturbance','Without disturbance')
                                 % try moving the legend with the mouse
grid;                            % add a  grid
hold off;
```

Closed-loop proportional control

We will first find the transfer function from reference to output:

$T_{RY}(s) = \dfrac{600K}{s + (60 + 600K)}$ (derive it, and provide a block diagram).

Therefore, when $W(s) = 0$ the output is given by $Y(s) = T_{RY}(s)R(s)$.

Also, the transfer function from disturbance to output is

$T_{WY}(s) = \dfrac{-1500}{s + (60 + 600K)}$ (derive it, and provide a block diagram).

Compute the sensitivity of the closed-loop transfer function $T_{RY}(s)$ with respect to changes in changes in controller gain K. How does it compare with $T_{WY}(s)$?

When $R(s) = 0$, the output due to the disturbance is given by $Y(s) = T_{WY}(s)W(s)$.

Combining, if both reference and disturbance are present, the total output is

$Y(s) = T_{RY}(s)R(s) + T_{WY}(s)W(s)$.

Considering these results, how would you want to pick the control constant K? (Explain your rationale!)

We will consider two design choices, and compare their performance:

$$K_1 = 10$$
$$K_2 = 50$$

1) Plot the step responses for both controllers in one plot.

```
% verify motor transfer function

motor

% the two controller gains

K1 = 10;
K2 = 50;

% find the transfer function from R(s) to Y(s) for each controller gain

T1_ry = feedback(K1*motor,1)          % the 1 indicates unity feedback

T2_ry = feedback(K2*motor,1)

% find the transfer function from disturbance W(s) to output Y(s)

T1_wy = feedback(motor,K1)*(-1500/600)

T2_wy = feedback(motor,K2)*(-1500/600)

% Plot the response to a step of magnitude 100 without disturbance

clf;                                % clear figure

step(100*T1_ry);                    % for controller K1
title('Closed-loop response to a step of magnitude 100')
hold on;
step(100*T2_ry);                    % for controller K2
legend('Using K_1','Using K_2');
hold off;
grid;
```

2) Find an approximation to the steady-state error for both designs from the plot.

```
zoom on;
% click on the plot with the mouse to find the approximate error value
zoom off;
```

3) Plot the response to a unit step disturbance for both designs.

```
% Consider a disturbance step input, with R(s) = 0
```

```
step(T1_wy);
hold on;
step(T2_wy);
title('Closed-loop response to a step disturbance');
legend('Using K_1','Using K_2');
hold off;
grid;
```

4) Plot the closed-loop response when both reference and disturbance are present.

```
% Compute the overall response, when R(s)=100/s and W(s)=1/s

t = [0:1e-6:1e-3];

y1_ref = step(100*T1_ry,t);
y2_ref = step(100*T2_ry,t);

y1_dist = step(T1_wy,t);
y2_dist = step(T2_wy,t);

y1 = y1_ref + y1_dist;
y2 = y2_ref + y2_dist;

plot(t,y1,'r')                   % plot response using K1 in red
hold on;
plot(t,y2,'b')                   % plot response using K2 in blue
legend('Using K_1','Using K_2')
grid;
xlabel('Time')
ylabel('Motor speed')
title('Closed-loop response to a step reference of magnitude 100 and a
unit step disturbance');
hold off;
```

Part II

Consider the plant

$G(s) = \frac{1}{(s+1)(s+2)}$.

1) What is the plant's type?

2) Let $C(s) = K$ (a proportional controller). Find the closed-loop transfer function from reference to output using unity feedback.

3) Choose different gains for K within the range 1 to 100. Plot the unit step response for the different gains. What happens with the transient response of the closed-loop as K increases?

4) For $K = 20$ find the maximum value attained by the output $y(t)$ and the settling time T_s for a unit step input (the time it takes the output to settle within a band

of $\pm 2\%$ around its final value). Also find what is the steady-state value of $y(t)$? What is the steady-state error equal to?

5) Design a controller that will increase the system's type by 1 and that will yield the smallest settling time you can obtain for a step input. What is the settling time? What is the steady-state error?

6) Plot the response of the closed-loop system to a unit ramp using the controller you designed in part (5).

   There are several ways to do this. One is to use the `sawtooth` command together with the `lsim` command to obtain the time response of the system. Another way is to implement the whole thing in Simulink, using a signal generator block to produce the ramp input. And a third way still is to use the `step` command yet again, noting that the Laplace transform of a unit step is $\frac{1}{s}$, and the Laplace transform of a unit ramp is $\frac{1}{s^2} = \left(\frac{1}{s}\right)\left(\frac{1}{s}\right)$.

   Record your observations. Is there a steady-state error? If so, what is its magnitude?

   What would happen with the steady-state error if the input were, instead, a train of steps? You do not need to give a plot for this question, just answer based on the type number of the plant together with your controller design.

7) Comment on the performance limitations you found in part (5). Do this by observing what happens if you make K very small, or very large.